Traffic-Aware Power-Gating Scheme for Network-On-Chip Routers

Nasim Nasirian*, Reza Soosahabi†, Magdy Bayoumi*
*The Center for Advanced Computer Studies, University of Louisiana, Lafayette, Louisiana 70504
†Distributed Computing & Communications Lab, Infolink-USA Inc., Lafayette, Louisiana 70508
Email: nxn5303@louisiana.edu, rsoosahabi@infolink-usa.com, mab@cacs.louisiana.edu

***Abstract*—Network-on-Chip technology has become very popular in the recent decade to address the scalability issue in multi-core processors. Developing the inter-connection network improves the performance of multi-core systems at the cost of additional power and design complexity. Power-gating of router buffers as the most power consuming part of Network-on-Chip has been proposed recently to reduce the static power consumption. However, power-gating has its own shortcomings in terms of delay and power overhead that can diminish the overall performance if it is not used in sync with the traffic flow. In this paper, a power management technique has been introduced to optimize the power-gating of router input buffers by utilizing an adaptive learning scheme to estimate the future traffic arrival. The proposed design intelligently invokes the power-gating signals to the router buffers based on the statistical estimation of the traffic inter-arrival times. Syncing the power-gating signals with the traffic arrival significantly reduces the power-gating overhead caused by: mandatory wake-up delay to route immediate traffic arrival and static power consumption due to prolonging idle state in anticipation of future arrival. The simulation results indicate that the proposed scheme has average 25% improvement in the static power consumption compared to previous works while it reduces the average packet latency by 35% in comparison to naive power-gating. It is worth mentioning that the overhead of power-gating has significantly decreased by 41% compared to the proposed scheme in [1].**

***Index Terms*—Network-on-Chip (NOC), Power-Gating(PG), Router, Routing Computation (RC), System-on-Chip (SOC).**

I. Introduction

In order to solve the scalability issue observed in bus-based interconnected system-on-chip (SOC), Network-on-chip (NOC) was introduced as novel approach, where each core/processor is a node that communicates to other nodes via designated routers. Similar to other VLSI designs, NOCs also suffer from power consumption and area limitation. Reaching to a legitimate tradeoff between the performance and the power consumption in NOC designs, has been the matter of research in [1]–[7]. Power-gating of idle buffers in routers is a popular solution to decrease the static power, but there are two main drawbacks in power-gating that should be addressed properly: (1) wake-up delay that is the time needed by a gated router buffer to resume its normal operation upon traffic arrival, which depends on it depends on the circuit parameters of the router. (2) Because of power overhead during sleep to active state transition, it is needed to assure that power-gating is beneficial. A very common solution is that to wait at least for T_{idle} cycles in idle state and then turn off the component if there was no incoming traffic but there will be static power consumption during T_{idle} until the buffer goes completely to sleep state. Additionally, there is still a chance to have an arrival shortly after turning off the buffer and it will totally degrade the efficiency of power-gating mechanism. An estimation for average wake-up delay, $T_{breakeven}$, the time needed to stay in sleep mode to compensate the wake-up power overhead, and T_{idle} have been calculated in [8]. These deficiencies in existing NOC power-gating approaches, inspired us to design an adaptive power-gating control unit that is based on future inter-arrival estimation. In this work, we have used the history of inter-arrival times as learning data to estimate the next inter-arrival time to efficiently implement the power-gating mechanism. This paper is organized as follows: in section II the proposed architecture is explained. In section III experimental results have been provided and section IV includes the conclusion.

II. Proposed Design

In this work a probabilistic approach has been proposed to optimize the power-gating of input buffers in NOC system. The proposed design consists of two main phases: (1), the system should be trained for the test database and traffics. Then the observation of idle time for every buffer in the routers should be collected. Depending on the router location in the network and routing algorithm, each port of router encounters different pattern of idle period. After data collection in the training phase, observed data are modeled as an estimated distribution function. (2), the decision for turning off the input buffers will be made based on the predicted inter-arrival time by the predictor. Fig.1 is representing the proposed router architecture for every port in a mesh topology. In the proposed design, the common router architecture in [6] has been employed. The additional block in the this architecture is the power-gating control unit. Predictor function and inter-arrival times monitor blocks are used in offline stage and can be removed in run time implementation. Fig. 2 is showing the state machine of the input buffers that should be updated by power-gating control unit. This unit in every router should monitor and control the power-gating mechanism for input buffers. Depending on the traffic of NOC and the type of application running on every core, traffic rate to the input buffers would be variant and dynamic.

978-1-5090-2757-6/16/$31.00

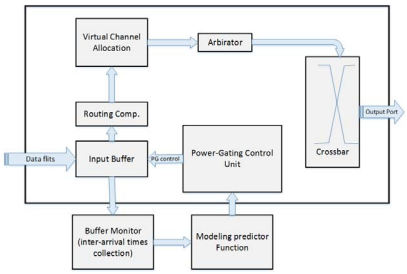


Fig. 1. Router Architecture.

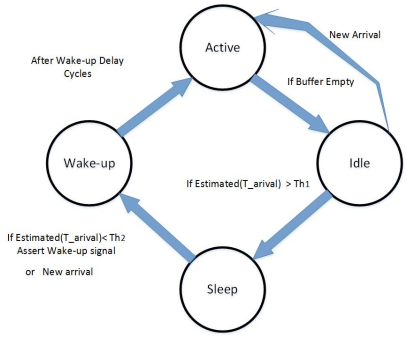


Fig. 2. State Diagram of Buffers in Power-Gating Control Unit.

Here four different power states have been defined for the input buffers: *active, idle, sleep* and *wake-up* states. In active state, buffers are working normally and in the idle state, there is no packet/flit waiting for the service. The wake-up state stands for the case that wake-up signal has been sent but due to the wake-up latency, the buffer is still not functional. Transition from idle state to sleep state will happen if the estimated idle period exceeds the $T_{breakeven}$ threshold (Th_1). Based on [8] it is assumed to be 8 clock cycles, otherwise it will stay in the idle state which is still consuming static power. If the new arrival occurs, the state of the port will be updated to active state immediately. By using the estimated Probability Density Function (PDF) of inter-arrival times, power-gating control unit will predict the probability of next arrival. Therefore, when the input port is in the idle state, it will define how much it is probable to have an arrival in the next $T_{breakeven}$ cycles. If the estimated probability was less than defined threshold, a sleep signal will be asserted to switching transistors of the input port.

The transition to wake-up state will happen whenever the control unit estimated value for the next arrival is less than wake-up delay cycles (Th_2) that is assume to be 9 clock cycles [8] or there was an incoming packet to the port. Apparently, if the estimation was not correct, it will wait in the sleep mode until receiving the incoming packet and then will force wake-up delay to the system.

In the training stage, the information of inter-arrival time will be collected as a training data set. After estimating the PDF of inter-arrival times, it can be used in the run-time phase to predict the future arrivals. The training phase can be repeated after observing performance degradation in the system implying that the application or traffic pattern have changed and it is needed to update the estimation parameters. In this phase, a naive NOC design has been executed and training data are collected from different packet injection rates (PIR). We have considered different spatial distributions: Butterfly, shuffle, bit-reversal and transpose. Arrival times into every input port have been recorded in order to estimate the inter-arrival distribution. We have assumed that the inter-arrival times of injected traffic by every local processing element into network are exponentially distributed.

III. Experimental Results

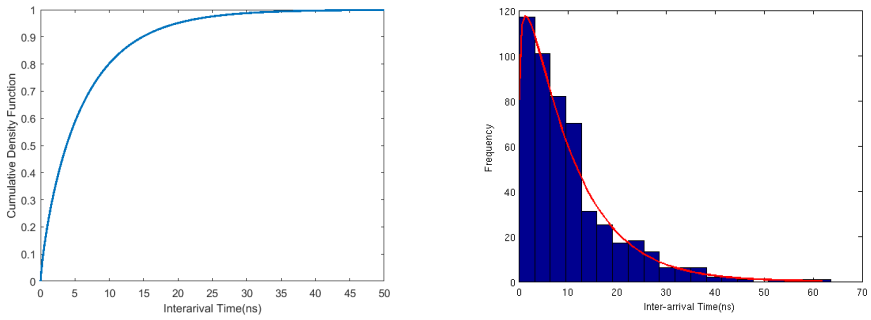
A cycle-accurate simulator (NOXIM) [9] has been used for simulation and collecting training data sets. We have considered two network sizes: 4×4 and 8×8 in a mesh topology. The Simulator is warmed up for 15000 cycles and each simulation runs for 150000 cycles for different PIR. As stated in [10], the accumulation of N arbitrary arrival processes will slowly converge to a Poisson process where the inter-arrival times are exponentially distributed for large N. One can observe that Gamma distribution will also converge to an exponential distribution where α in (1) tends to 1 as the network size grows. Since the inter-arrival time are the superpositions of all previous exponential queuing delays, a Gamma PDF [11] has been estimated in (1) for the inter-arrival times based on training data sets using MATLAB®:

$$f(x|\alpha,\beta) = \frac{x^{\alpha-1}}{\beta^{\alpha}\Gamma(\alpha)} e^{\frac{-x}{\beta}} \quad (1)$$

Where $\Gamma(.)$ denotes Gamma function and α and β are shape and scale parameters respectively [11]. The plots of Cumulative Density Function (CDF) and histogram of training data set has been shown in Fig. 3 (a) and Fig. 3 (b) respectively. Fig. 3 (a) shows the probability of having inter-arrival time less than or equal to the x-axis values. In Fig. 3 (b), the frequency of different inter-arrival times have been presented and it has been fitted to a Gamma distribution (red color curve). It shows that the most frequent inter-arrival time is less than $6nsec$.

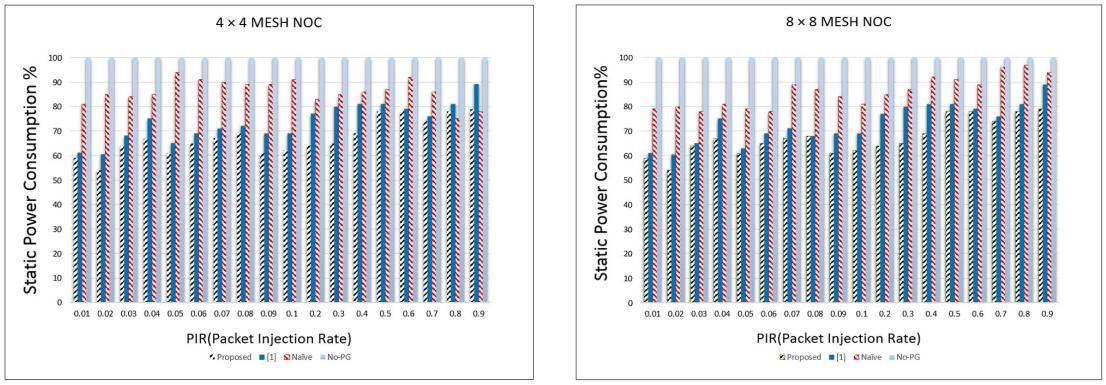
In the run-time phase, in order to meet the requirement of the proposed scheme, PG control unit has been added to the simulator. This unit performs asserting the wake-up and sleep signals to switch transistors of input buffers. The network spatial distribution of traffic is assumed as random to have a better evaluation of trained model. It means that destination of packets at every source will be selected randomly. Two network sizes of 4×4 and 8×8 2D-mesh with 10 router buffers per input have been simulated on different packet injection rates (PIR) with exponentially distributed inter-arrival times and the Gamma PDF parameters have been estimated based on training database as follows :
$\alpha = 1.148$, $\beta = 7.73529$ in 4×4 network ,
$\alpha = 0.87$, $\beta = 7.236$ in 8×8 network.

To analyze the power consumption precisely, we have performed hardware implementation of router using Synopsys®



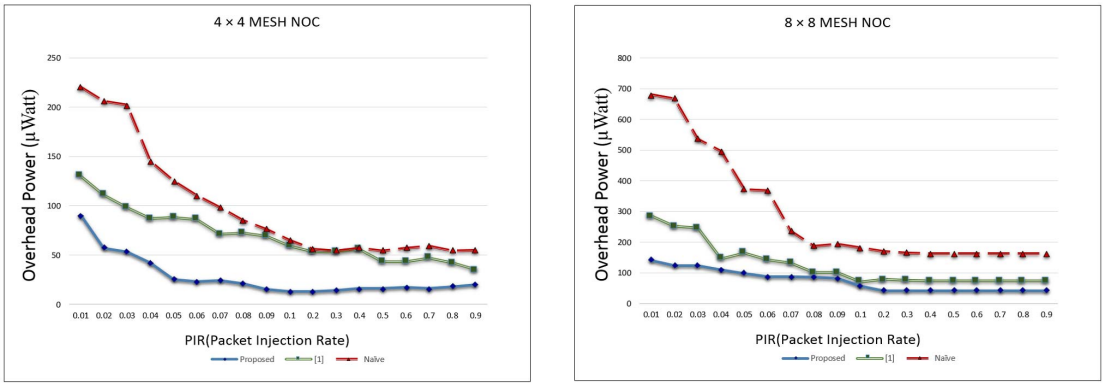
(a) (b)

Fig. 3. (a) Cumulative Density Function of Training Data $P(X \leq x)$. (b) Histogram of Inter-arrival Time.



(a) (b)

Fig. 4. Static Power Consumption(normalized to No-PG) under Random spatial distribution (a) 4×4 2D-Mesh and (b) 8×8 2D-Mesh

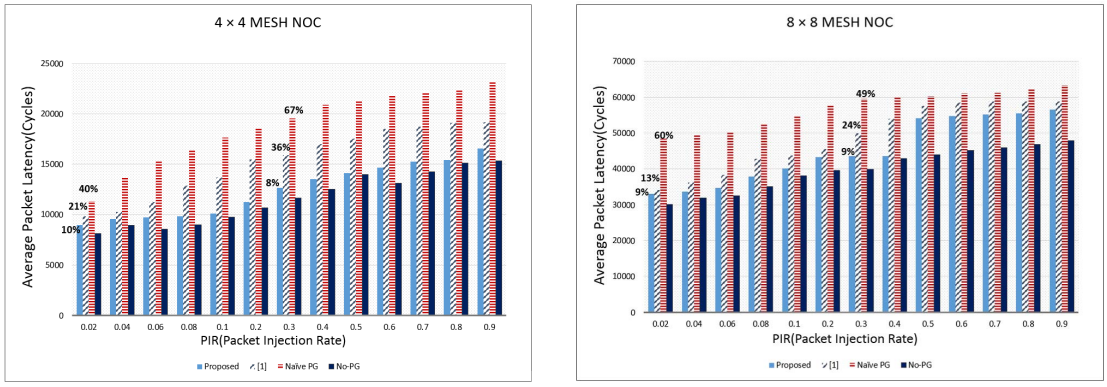


(a) (b)

Fig. 5. Power-gating overhead (dynamic power) under Random spatial distribution (a) 4×4 2D-Mesh NOC and (b) 8×8 2D-Mesh

$45nm$ technology in 500 MHz frequency and its static and dynamic power consumption results have been supplied into simulator for power estimation purposes. we have assumed the wake-up latency $= 9$ cycles, $T_{breakeven} = 8$ cycles and $T_{idle} = 4$ cycles [8]. For the comparison of proposed method, three other designs including naive PG, No-PG and [1] has been simulated along side the proposed scheme.

In Fig. 4 (a) and (b), the amount of static power consumption for the mentioned methods in a 4×4 and 8×8 2D-mesh have been presented. All values have been normalized to No-PG results. It can be seen, there is a significant improvement in overall static power consumption in the proposed method compared to No-PG and the naive power-gating method. The proposed design also showed near 10% reduction in static power consumption compared to the design in [1]. Since in the naive PG design, transition to sleep state will happen after a fixed cycles waiting in idle state, it imposes more static power to the system. having the intelligence to predict next arrival,



(a) (b)

Fig. 6. Average Packet Latency in cycles under Random spatial distributions (a) 4×4 2D-Mesh and (b) 8×8 2D-Mesh

TABLE I
HARDWARE IMPLEMENTATION RESULTS

45nmTech	Proposed Router	Regular Design
Area	4104 Cells	3556 Cells
Cell Leakage Power	$57.67uW$	$54.95uW$

power-gating can be performed more efficiently. Moreover, by predicting the next arrival in the sleep state and sending wake-up signals in advance, it also covers the wake-up delay.

Fig. 5 (a) and (b) depicts the overhead power consumption to the system for two different network sizes. This overhead is mainly related to the wake-up power that is inevitable in PG. In this figure, the main disadvantage of power-gating can be seen in the naive PG and [1]. By avoiding inefficient PG using the future arrival estimation, we could reduce the power-gating overhead significantly by 37% and 46% compared to [1] in 4×4 and 8×8 networks respectively.

It is worth to note that the average packet latency is also reduced in the proposed design. Fig. 6 (a) and (b) exhibits the comparison between different power-gating scheme and No-PG design. As it is expected, the naive PG has the worst latency because of experiencing wake-up delays. Although this wake-up latency has been compensated partially in [1], it still has non-negligible delay. Our proposed design could achieve near 24% less latency compared to [1] and 48% to naive PG scheme in 4×4 network, and in the 8×8 network, the improvement is 12% compared to [1] and 33% compared to naive PG respectively.

To evaluate the area overhead of proposed scheme, the router micro-architectural implementation results have been illustrated in Table I alongside of regular router design [6] using Synopsys® $45nm$ technology. It can be seen that area has been increased by 15% in the proposed router that is not significant cost to achieve the remarkable power-saving.

IV. CONCLUSION

In pursuance of decreasing the power-gating penalties in NOC designs, a stochastic approach has been employed to manage the power-gating decisions. By using the training data set, probability density function of inter-arrival times for every input has been estimated. The proposed design avoids the uncompensated power-gating and covers the wake-up delay by asserting wake-up signals in advance by predicting the next arrival. Our simulation results have shown improvement in the average packet latency compared to previous works and also noticeable overhead reduction accompanied with equal or better static power saving.

REFERENCES

[1] H. Matsutani, M. Koibuchi, H. Amano, and D. Wang, "Run-time power gating of on-chip routers using look-ahead routing," in *Design Automation Conference, 2008. ASPDAC 2008. Asia and South Pacific*. IEEE, 2008, pp. 55–60.

[2] M. R. Casu, M. K. Yadav, and M. Zamboni, "Power-gating technique for network-on-chip buffers," *Electronics Letters*, vol. 49, no. 23, pp. 1438–1440, 2013.

[3] L. Chen, D. Zhu, M. Pedram, and T. M. Pinkston, "Power punch: Towards non-blocking power-gating of noc routers," in *High Performance Computer Architecture (HPCA), 2015 IEEE 21st International Symposium on*. IEEE, 2015, pp. 378–389.

[4] R. Parikh, R. Das, and V. Bertacco, "Power-aware nocs through routing and topology reconfiguration," in *Design Automation Conference (DAC), 2014 51st ACM/EDAC/IEEE*. IEEE, 2014, pp. 1–6.

[5] N. Nasirian and M. Bayoumi, "Low-latency power-efficient adaptive router design for network-on-chip," in *System-on-Chip Conference (SOCC), 2015 28th IEEE International*, Sept 2015, pp. 287–291.

[6] W. J. Dally and B. P. Towles, *Principles and practices of interconnection networks*. Elsevier, 2004.

[7] H. Jiang, M. Marek-Sadowska, and S. R. Nassif, "Benefits and costs of power-gating technique," in *Computer Design: VLSI in Computers and Processors, 2005. ICCD 2005. Proceedings. 2005 IEEE International Conference on*. IEEE, 2005, pp. 559–566.

[8] Z. Hu, A. Buyuktosunoglu, V. Srinivasan, V. Zyuban, H. Jacobson, and P. Bose, "Microarchitectural techniques for power gating of execution units," in *Proceedings of the 2004 international symposium on Low power electronics and design*. ACM, 2004, pp. 32–37.

[9] F. Fazzino, M. Palesi, and D. Patti, "Noxim: Network-on-chip simulator," *URL: http://sourceforge. net/projects/noxim*, 2008.

[10] R. Banys, "Weak convergence of the sum of independent point processes to a poisson process," *Lithuanian Mathematical Journal*, vol. 17, no. 1, pp. 11–16, 1977.

[11] P. G. Hoel *et al.*, "Introduction to mathematical statistics." *Introduction to mathematical statistics.*, no. 2nd Ed, 1954.